SUBSTITUTE SPECIFICATION WITH EDITS MARKED THEREON

# MESSAGE CHAINING TRANSMISSION PROCESS AND SYSTEM FOR DATA BASES.

## Introduction

The present invention concerns a ~~process and~~ method and a system ~~of database~~ for updating databases, ~~and~~ in particular during the transmission of a chain of messages.

## Background of the Invention

In a system comprising a management centre and a plurality of subscribers over a wide territory, it is known to send ~~the sending~~ by telephone or ~~hertz route of~~ terrestrial broadcast updating information for the database of these subscribers ~~is known~~. These messages ~~are~~ may be addressed~~,~~ either to all subscribers, or to one subscriber in particular~~, that is to say that it they contains a subscriber module address~~.

These messages are intended for ~~the administration of~~ administering the system and are themselves superposed to the useful data such as video, audio or data. ~~One understands that the place used by these messages is limited. Another limitation applies to t~~The length of the message ~~which~~ is therefore limited by the fact that the useful data can be interrupted only for a short moment. ~~One understands that~~For example, in the ~~example~~ case of an audio/video transmission, the emission channel can be interrupted only for a short moment so that no visual impact will be perceptible.

This is why, for ~~transmission of~~ transmitting a large amount of data, it was found necessary to divide them in a large number of messages.

These messages are sent in sequence on the network, in a logic order, that is to say one after the other, separated by a short interval, for example one second.

As certain systems of this type do not use a return channel towards the ~~managing~~ management centre, i.e. a channel enabling a communication from the subscriber towards the management center, as for example a modem, it is difficult for the ~~managing~~ management centre to know if the sent data ~~sent is~~ has arrived correctly. ~~In this way~~Accordingly, it is ~~obliged~~ necessary to repeat these messages periodically so as to statistically ensure that each message ~~is~~ has arrived at its destination.

A subscriber module includes schematically a digital receiver, either audio~~,~~ or video or data~~,~~ (~~and~~ or possibly a combination of these three types), and a decoder able to extract the management messages, these latter being directed towards a security module comprising the subscriber database. This security module can be directly installed in the subscriber module or, for security and cost reasons, it can be realized as a ~~detachable~~ removable module such as ~~an~~ smart card or microchip card.

The messages arriving to the security module are processed by ~~the~~ a command interpreter. It is possible that the messages do not arrive in the ~~broadcasting~~ order in which they were broadcast, either because of interferences in the transmission or simply as the subscriber~~'s~~ unit was not switched on when ~~at the moment of the sending of~~ previous messages were sent. It is therefore necessary to specify that prior ~~each~~ to processing, each message is first decrypted and controlled for its authenticity. A message which does not satisfy the control criteria is rejected. ~~In this case,~~If the security module ~~will~~ receives ~~for example~~ the third message before the first and second messages~~.~~, ~~T~~the execution of the third message without the prerequisite execution of two previous messages can lead to blocking ~~of~~ the database or ~~to~~ may generate an error.

A first solution consists in memorising all the messages constituting a chain and, when it is complete, to go on to its processing. This solution has the drawback to ~~set~~ limiting the length of the maximum chain according to the available memory ~~available~~.

Since ~~T~~the memory capacity of ~~detachable~~ removable smart cards is limited, ~~which~~ this obliges the card to process each message when they arrive.

Brief Summary of the Invention

The problem that the present invention proposes to solve, is to suppress on the subscriber's database the ~~disastrous~~ harmful effects due to the execution of messages in an order different ~~to~~ from that initially foreseen.

This aim is fully reached by a ~~transmission process of a chain of subscriber's database management messages, this process consisting of~~ method of transmission of a chain of database management messages, the method comprising the step of associating with each message a conditional block which determines if the current message is to be processed with reference to all or part of others ~~element~~ message members of the chain and, in the negative event, the conditional block ~~comprises the~~ specifies conditions ~~bound~~ linked to ~~the~~ a previous processing of all or part of other ~~elements~~ messages member of the chain.

In fact, ~~due~~ thanks to this new conditional block included in each message ~~elements~~ member of a chain, it is possible to determine if this message can be processed separately or if it must satisfy ~~the~~ processing conditions related to ~~of processing of the~~ messages supposed to be received previously. ~~It is obvious that t~~This test also allows ~~also~~ the system to determine if the presently evaluated message has already been processed ~~the determination if the current message evaluation has already been processed~~.

To reach this aim, the security module disposes of a memory organized under the form of table indicating, for each chain, which ~~are the~~ messages members of ~~this~~ the chain ~~that~~ have already been processed. After having ~~processing~~ processed ~~of~~ all the members ~~elements~~ of the chain, the table of this chain is maintained in order to avoid that the resending of the same chain restarts its execution. It can be deleted on request by the management centre or after a predefined time.

The conditional block contained in the message does not only contain ~~only~~ a simple indication binding the processing of the current message to the condition of having executed the ~~carried out the execution of~~ previous messages, but also covers more

complex functions, such as conditions related to each ~~element~~ member of the message chain. For example, it is possible to ~~define~~ bind the processing of element 4 of the chain ~~on~~ to the condition that either element 1 or 2 is processed and that element 3 is imperatively processed. We will thus have the function:

F(4) = (1 or 2) and 3.

~~We take~~In the example of the arrival ~~to~~ at the security module of a message member of family 5, this message being the ~~element~~ member 4 of this family,~~.~~ ~~T~~the first operation will be to determine if its processing is ~~subject~~ bound to other conditions. If this is not the case, it can be processed immediately. It should be noted that ~~to chain~~ the fact of chaining messages does not mean that the processing must be made in the index order of the chain. One can imagine the case where one loads a bulky software, and for this reason, one divides it to transmit it in a chain of messages. Each of these messages contains a loading address and the corresponding data. This is why an element of the chain can be processed in an indifferent order. On the other hand, the last ~~element~~ member of the chain setting up this new software will contain a condition stating that all the ~~elements~~ members of the chain must have been ~~carried out~~ executed in order to enable the execution of ~~that~~ this software ~~can be executed~~. When this condition ~~has been~~ is satisfied, the table ~~correspondent~~ corresponding to this family indicates that all messages have been ~~carried out~~ executed.

According to a variant of the invention, the conditional block is divided in two parts, ~~the~~ one being called "operation" to describe the type of logic function and the other being called "related ~~element~~ member" to describe ~~on~~ which other ~~elements~~ members the operation must apply. The format of the part "related ~~element~~ member" corresponds to the format used in the table stored in the database designating the state of processing of the ~~elements~~ members of the chain. In this way, the logic comparison is greatly facilitated.

According to other embodiments, the conditional block does not refers ~~not~~ to all the other ~~elements~~ members of the chain, but only to some ~~only~~ of them. It would be ~~for example~~ possible to refer~~r~~ to three previous elements and not to all the elements. This allows the reduction of the length of the conditional block and takes into account the fact that an interference rarely exceeds the time of three messages. According to another example, one could define a chain structure where only the last element contains a conditional block.

This structure allows, unlike the solutions of the prior art, the system to reject only a minimum number of messages. ~~In fact~~Previously, when a message ~~is missed~~ was missing in a chain, all the following messages were rejected until the ~~new passage~~ receipt of the missing message. The execution of a chain was ~~in this way~~ thus dependent upon the continuous reception of the members ~~elements~~ of the chain, each missing element ~~missing~~ leading to the rejection of all messages having a~~n~~ higher index than the missing message.

According to an embodiment of the invention, the subscriber module, besides sending the messages to the security module, includes a memory to memorise them as soon as they arrive.

Therefore, it is possible that the absence of a message containing a condition on a ~~preceding~~ previous message leads to reject all the following messages. When this awaited message arrives, it is of course processed authorising the processing of the other messages. It is possible otherwise that a long time elapsed before these missing messages are present in the transmission with the risk that some are rejected, for example due to the bad quality of the connection between the managing centre and the subscriber module.

To minimize the number of repeated messages necessary for the completion of the chain, the security module can accede to the memory located in the subscriber module since it contains all the messages in their arrival order. Thus, as soon as the missing message arrives and its processing completed, the security module asks the reading of the memory

to process all the messages which have been rejected because of the condition on the missing message.

An important aspect of the invention lies in presenting each message to the security module while storing it in a memory in the subscriber module. This principle can include exceptions when some messages are not ~~destined~~ intended to the security module but only to the subscriber module~~of the subscriber~~. Thus, ~~even if~~ although some messages are rejected by the security module as the conditions are not fulfilled, this ~~system~~ module knows that this message is contained in the memory of the subscriber module and can, when the condition is fulfilled, accede to ~~this~~ the memory to ~~proceed~~ process these messages instead of awaiting a next passage of the following messages.

In an embodiment, the memory of the subscriber module is organised as a stack with entry in series, each new entry causing the displacement of the previous entry (first-in first-out).

The reading by the security module can be realised in different ways. It can ask the transmission of an exact address of the memory. Nevertheless, an important aspect of the security in this kind of application lies in the confidentiality of the organisation of the data. For this reason, instead of asking the transmission of a specific address, the security module asks the subscriber module to submit all or part of the messages contained in its memory. It is the task of the security module to sort out between the messages already ~~carried out~~ processed and the messages to ~~carry out~~be processed.

Brief description of the drawings

The invention will be better understood based on the following detailed description which refers to ~~annexed~~ the enclosed drawings which are given by way of a non limitative example, wherein:

- Figure 1 represents a message sent according to the systems of the prior art;
- Figure 2 represents a message sent according to the invention;

- Figure 3 represents one embodiment for ~~updating~~ implementing the temporary memory of the subscriber module.

Detailed description of the invention

In Figure 1 the different blocks of a message which take part in the function of chaining are represented schematically. We find a first header block HD, which describes the kind of message, and contains the information that this message is part of a chain. To form the chain, a second family block FM indicates to which family this message belongs. In fact, it is possible that several chains are transmitted simultaneously and in this case the identification of the family is necessary. Now that the family is defined, the subsequent block FI is used to identify each element of the family and its place in the chain. So, with these two data, each element of the family can be placed at the right place in the chain. It is known to indicate either in control block F1 or in control block FM ~~in one or the other of control blocks FI or FM~~ the maximum number of ~~the element~~ members of the family. This function can equally be obtained by a particular marking of the last element of the family.

In the example of figure 2, ~~in~~ a conditional block CD is added to the message of Figure 1~~, showing the two blocks FM and FI, one add a supplementary~~ Conditional block CD ~~which~~ determines ~~a~~ the conditions necessary to ~~carry out~~ execute this message. According to a first embodiment of the invention, this block is constituted by a bit which indicates if ~~the~~ a previous message ~~should~~ must ~~or should not~~ have been executed. If this condition is requested, the interpreter in charge of the operations on the database~~,~~ will verify ~~if~~ whether the previous message has been executed ~~properly~~ and in the positive event, will execute this new message.

In another embodiment, this conditional block CD ~~is constituted by~~ comprises a field comprising groups, a group for each element of the chain. Each group contains a condition on an element of the chain and can have several meanings, for example the

condition "must have been executed", "can ~~be~~ have been executed" or "must not have been ~~be~~ executed". The latter condition is generally ~~is~~ the ~~complement~~ mirror of the first.

~~We take~~ In the example of a chain of 6 elements, ~~the~~ in which element 3 ~~should~~ must imperatively be ~~carried out~~ executed before element 5~~. In this case~~, one can specify in message 3 that it ~~should~~ must not be ~~carried out~~ executed if message 5 was not ~~processed~~executed. This condition can lead to a blocking if one does not specify the inverted condition in message 5. In this case, message 5 will contain the condition "must ~~be~~ have been executed"~~,~~ in reference to the message 3~~, in order~~ so that if message 5 arrive~~d~~s before message 3, it will ~~be~~ not be processed.

In Figure 3, an implementation of the memory M of the subscriber module and the connection with the security module are represented. The incoming ~~flux~~ stream is firstly filtered by a module SEL, which has the task ~~to separate~~ of separating the managing messages from the other data. These messages are then transmitted to the selection module SW which has the task ~~to~~ of sending them to the different modules i.e. the security module SM, ~~to~~ the processing centre CTR of the subscriber module STB and ~~to~~ the memory M of the subscriber module. The storage into the memory of these messages causes the increment of the input message pointer so that no message will be lost, the oldest message being then eliminated from the memory. In the same time~~way~~, these messages are transmitted to the security module, represented here as a~~n~~ smart card SM. This card SM contains a first memory managing module GM and a ~~control~~ command interpreter INT for managing the ~~controls~~ commands of the database BD. This memory manager GM can dialogue with the processing centre CTR by the connection I/O and by this means, to influence the connections in the selection module SW. The dotted line represented in Figure 3 represents the subscriber module STB. All the ~~managing~~ management messages addressed to the security module SM are directed by the selector SW to the security module, in particular to the memory manager GM, and are then ~~are~~ transmitted to the ~~control~~ command interpreter if the processing conditions are fulfilled. The memory manager GM updates the table of the processed messages ~~processes~~ to make the necessary comparisons at the moment of the arrival of a new message. The

connection with the smart card SM is of in/out type and ~~in this way~~ thus information and controls commands can be sent ~~at destination of~~ to the subscriber module, this connection being represented by the line I/O.

As explained previously, the memory M is physically in the subscriber unit STB. This is why the card SM can, ~~by the intermediate of the line I/O~~ via the I/O line, ask the availability of a memory section so as to be able to store the messages of a chain. In our example, the maximum number of elements in a chain does not exceed 16. ~~So~~Thus, at the arrival of the first element of the chain, the card SM, ~~by~~ via the line I/O, requests the reservation of at least 16 memory places. If, during the transmission of this first chain, another chain is announced, the card will ask the reservation of 16 new places in order to ensure the storage of a maximum number of members of the chain according to the receiving conditions.

In order to read the data contained in the memory M, for example the position M3, the card SM can order, through the selection module SW, the address multiplexer AMUX to return the content of this memory position. In order to forward these data towards the card, a data multiplexer DMUX has the function ~~to~~ of reading the required memory position ~~required~~ and ~~to transfer~~ of transferring it towards the card. These different transfers are directed by the selection module SW.

When the processing of the chain has been interrupted due to an interference on a message for example, the other messages continue to be stored in the memory of the subscriber module. When the missing message is retransmitted by the managing centre, it is executed properly and the memory manager GM recalls all the other messages of the chain ~~acceding~~ by accessing the memory of the subscriber module. In this case, the entry of the smart card SM is not ~~made~~ any longer made on the arrival of messages but on the contents of the memory M. This access to memory M can either be made in direct access specifying a memory address, or by sequential access by reading the messages in their arrival order.

In ~~an~~ one embodiment, the memory M is organized as a buffer memory ~~buffer of~~ having a fixed length according to the availability of the free memory of the subscriber module. This memory includes an input pointer ~~increased~~ incremented on each message introduced in the memory, and an output pointer ~~increased~~ incremented on each reading by the memory manager GM.

The communication possibility between the card SM and the subscriber module STB, in particular the centre CTR, authorises more complex functions. One of the ~~problem~~ frequently met problems at the moment of the replacement of one or the other of the elements of the system, either the card or the subscriber module, is to ensure the compatibility of the functions with the material of previous generations. For this, it is interesting to allow communication between the different elements in order to establish the functions available in each of them; this is the task of the ~~line~~ I/O line which allows ~~to send~~ sending instructions ~~of~~ from the card to the subscriber module. These instructions can, for example, ask the subscriber module to communicate its audio, video or data functions, the generation of the module or the software version. To answer to this request, the module STB disposes of means to compose a managing message and to transmit it~~,~~ either in the memory M for further reading by the card, or directly to the card, such as represented in Figure 3.

According to another embodiment of the invention, the module STB disposes of a ~~connection by~~ modem connection with the managing centre. In this case, the announcement of resources can be made by the module STB to the managing centre through the modem, on request of the security module SM.

As indicated in Figure 3, the module STB also receives ~~in the same way~~ the managing messages coming from the managing centre. The messages arriving to the processing centre CTR can contain a configuration request instruction. The response can be made by the modem connection or be transmitted to the card SM. Some of these managing messages are only destined to the module STB and the processing centre CTR,

responsible ~~to~~ from the management of the module STB, will not transmit them to the security module SM or to the memory M.

## ABSTRACT

~~This invention consists in a process and in a transmission system of~~ The processing of a chain of database updating messages transmitted between a managing centre and a plurality of subscriber databases is improved by ~~geographically shared. Each message includes a chain identifier (FM) and a chain index (FI) allowing the identifcation of the message in the chain. If a message is not received following an interference in the connection, the processing of further messages can cause the locking of databases. In order to avoid this drawback, the solution consists in~~ adding to each message a condition block (CD) which determines ~~if this~~ whether that message ~~has to~~ may be processed without reference to other elements of the chain, and if not, for defining ~~or which are~~ the conditions that must be linked to the ~~previous~~ processing of that elements of the chain.